Distributed Self-triggered Control for Multi-agent Systems

Dimos V. Dimarogonas, Emilio Frazzoli and Karl H. Johansson

***Abstract*— It is desirable to limit the amount of communication and computation generated by each agent in a large multi-agent system. Event- and self-triggered control strategies have been recently proposed as alternatives to traditional time-triggered periodic sampling for feedback control systems. In this paper we consider self-triggered control applied to a multi-agent system with an agreement objective. Each agent computes its next update time instance at the previous time. This formulation extends considerably our recent work on event-based control, because in the self-triggered setting the agents do not have to keep track of the state error that triggers the actuation between consecutive update instants. Both a centralized and a distributed self-triggered control architecture are presented and shown to achieve the agreement objective. The results are illustrated through simulated examples.**

I. INTRODUCTION

Distributed control of networked multi-agent systems is an important research field due to its role in a number of applications, including multi-agent robotics [6], [16], [8], [3], distributed estimation [20],[23] and formation control [12],[5],[28], [2],[22].

Recent advances in communication technologies have facilitated multi-agent control over communication networks. On the other hand, the need to increase the number of agents leads to a demand for reduced computational and bandwidth requirements per agent. In that respect, a future control design may equip each agent with a small embedded micro-processor, which will collect information from neighboring nodes and trigger controller updates according to some rules. The control update scheduling can be done in a time-driven or an event-driven fashion. The first case involves the traditional approach of sampling at pre-specified time instances, usually separated by a specific period. Since our goal is allowing more agents into the system without increasing the computational cost, an event-driven approach seems more suitable. Stochastic event-driven strategies have appeared in [21],[14]. Similar results on deterministic event-triggered feedback control have appeared in [26],[24],[13],[15],[11]. A comparison of time-driven and event-driven control for stochastic systems favoring the latter can be found in [4].

Motivated by the above discussion, in previous work [7] a deterministic event-triggered strategy was provided for a large class of cooperative control algorithms, namely those that can be reduced to a first order agreement problem [19]. In contrast to the event-triggered approach, we consider in this paper a self-triggered solution to the multi-agent agreement problem. In particular, each agent now computes its next update time at the previous one, without having to keep track of the state error measurement that triggers the actuation between two consecutive update instants. The approach is first presented in a centralized fashion, while a distributed counterpart is presented next. Self-triggered control is a natural extension of the event-triggered approach and has been considered in [25],[1],[27],[17],[18].

The rest of this paper is organized as follows: Section II presents some necessary background and discusses the problem treated in the paper. The centralized case is discussed in Section III where we first review the event-triggered formulation of [7] and proceed to present the self-triggered approach of the current paper. Section IV presents the distributed counterpart, first reviewing the results of [7] and then presenting the distributed self-triggered framework. Some examples are given in Section V while Section VI includes a summary of the results of this paper and indicates further research directions.

II. PRELIMINARIES

A. System Model

We consider N agents, with $x_i \in \mathbb{R}$ denoting the state of agent i. Note that the results of the paper are extendable to arbitrary dimensions. We assume that the agents' motion obeys a single integrator model:

$$\dot{x}_i = u_i, \qquad i \in \mathcal{N} = \{1, \ldots, N\}, \tag{1}$$

where u_i denotes the control input for each agent.

Each agent is assigned a subset $N_i \subset \mathcal{N}$ of the rest of the team, called agent i's *communication set*, that includes the agents with which it can communicate. The undirected *communication graph* $G = \{V, E\}$ of the multi-agent team consists of a set of vertices $V = \{1, ..., N\}$ indexed by the team members, and a set of edges, $E = \{(i,j) \in V \times V | i \in N_j\}$ containing pairs of vertices that correspond to communicating agents.

B. Background and Problem Statement

The agreement control laws in [9], [19] were given by

$$u_i = -\sum_{j \in N_i} (x_i - x_j), \tag{2}$$

and the closed-loop equations of the nominal system (without quantization) were $\dot{x}_i = -\sum\limits_{j \in N_i} (x_i - x_j,), \ i \in \mathcal{N}$, so

Dimos V. Dimarogonas and Emilio Frazzoli are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, U.S.A. {ddimar,frazzoli@mit.edu}. Karl H. Johansson is with the KTH ACCESS Linnaeus Center, School of Electrical Engineering, Royal Institute of Technology (KTH), Stockholm, Sweden {kallej@ee.kth.se}. The work of the third author was supported by the Swedish Research Council (VR), the Swedish Governmental Agency for Innovation Systems (VINNOVA) and the Swedish Foundation for Strategic Research (SSF),

that $\dot{x} = -Lx$, where $x = [x_1, \ldots, x_N]^T$ is the stack vector of agents' states and L is the Laplacian matrix of the communication graph. For a review of the Laplacian matrix and its properties, see the above references and [10]. For a connected graph, all agents' states converge to a common agreement point which coincides with the average $\frac{1}{N}\sum_i x_i(0)$ of the initial states.

We redefine the above control formulation to take into account event-triggered strategies for the system (1). Both centralized and distributed event-triggered cooperative control are treated.

1) Centralized Event-triggered Multi-agent Control: For each $i \in \mathcal{N}$, and $t \geq 0$, introduce a (state) measurement error $e_i(t)$. Denote the stack vector $e(t) = [e_1(t), \ldots, e_N(t)]^T$. The discrete time instants where the events are triggered are denoted by: $t_0, t_1, \ldots$. To the sequence of events $t_0, t_1, \ldots$ corresponds a sequence of control updates $u(t_0), u(t_1), \ldots$ Between control updates the value of the input u is held constant and equal to the last control update, i.e.,:

$$u(t) = u(t_i), \forall t \in [t_i, t_{i+1}), \tag{3}$$

and thus the control law is *piecewise constant* between the event times $t_0, t_1, \ldots$. The centralized cooperative control problem is stated as follows: "derive control laws of the form (3) and event times $t_0, t_1, \ldots$ that drive system (1) to an agreement point equal to their initial average."

2) Distributed Event-triggered Multi-agent Control: In the distributed case, there is a separate sequence of events $t_0^k, t_1^k, \ldots$ defined for each agent k. A separate distributed condition triggers the events for agent $k \in \mathcal{N}$. The distributed control law for k is updated both at its own event times $t_0^k, t_1^k, \ldots$, as well as at the last event times of its neighbors $t_0^j, t_1^j, \ldots, j \in N_k$. Thus it is of the form

$$u_k(t) = u_k(t_i^k, \bigcup_{j \in N_k} t^j_{i'(t)}), \tag{4}$$

where $i'(t) \triangleq \arg \min_{l \in \mathbb{N}: t \geq t_l^j} \left\{ t - t_l^j \right\}$.

The distributed cooperative control problem can be stated as follows: "derive control laws of the form (4), and event times $t_0^k, t_1^k, \ldots$, for each agent $k \in \mathcal{N}$ that drive system (1) to an agreement point equal to their initial average."

III. Centralized Self-triggered Control

We now present a self-triggered control design for the agreement problem. The event-triggered formulation of [7] is reviewed first and it is then modified to the self-triggered design.

A. Review of Centralized Event-Triggered Control Design

The state measurement error is defined by

$$e(t) = x(t_i) - x(t), \qquad i \in \mathcal{N}, \tag{5}$$

for $t \in [t_i, t_{i+1})$. The choice of t_i will be given in the sequel. The proposed control law in the centralized case is defined as the event-triggered analog of the ideal control law:

$$u(t) = -Lx(t_i), \qquad t \in [t_i, t_{i+1}) \tag{6}$$

The closed loop system is then given by

$$\dot{x}(t) = -Lx(t_i) = -L(x(t) + e(t)) \tag{7}$$

Denote by $\bar{x}(t) = \frac{1}{N}\sum_i x_i(t)$ the average of the agents' states. It is shown in [7] that $\bar{x}(t) = \bar{x}(0) = \dfrac{1}{N}\sum_i x_i(0) \equiv \bar{x}$, i.e., the average of the agents' states remains constant and equal to its initial value.

Thus in the event-triggered set up of [7], the event times t_i, $i = 0, 1, \ldots$ are defined recursively by

$$t_{i+1} = \arg\min_t \{t : \|e(t)\| = \sigma \frac{\|Lx(t)\|}{\|L\|}, t \geq t_i\} \tag{8}$$

with $t_0 = 0$. This also implies that the condition

$$\|e\| \leq \sigma \frac{\|Lx\|}{\|L\|}, \tag{9}$$

holds for all times and that the control is updated when this condition is violated. The main result of [7] is summarized in the following:

Theorem 1: Consider system $\dot{x} = u$ with the control law (6),(8) and assume that the communication graph G is connected. Suppose that $0 < \sigma < 1$. Then the state of all the agents converge to their initial average, i.e., $\lim_{t\to\infty} x_i(t) = \bar{x} = \frac{1}{N}\sum_i x_i(0)$ for all $i \in \mathcal{N}$.

B. Self-triggered Control

In the event-triggered formulation, it becomes apparent that continuous monitoring of the measurement error norm is required to check condition (8). In the context of self-triggered control, this requirement is relaxed. In contrast, in the self-triggered setup, the next time t_{i+1} at which control law is updated is predetermined at the previous event time t_i and no state or error measurement is required in between the control updates. Such a self-triggered control design is presented in the following.

For $t \in [t_i, t_{i+1})$, (7) yields $x(t) = -Lx(t_i)(t - t_i) + x(t_i)$. Thus (9) can be rewritten as $\|x(t) - x(t_i)\| \leq \sigma \frac{\|Lx(t)\|}{\|L\|}$, or

$$\| - Lx(t_i)(t - t_i)\| \leq \sigma \frac{\| - L^2 x(t_i)(t - t_i) + Lx(t_i)\|}{\|L\|},$$

or, equivalently

$$\|Lx(t_i)\|(t - t_i) \leq \frac{\sigma}{\|L\|}\|(-(t - t_i)L + I)Lx(t_i)\|.$$

An upper bound on the next execution time t_{i+1} is given by

$$\|Lx(t_i)\|(t^* - t_i) = \frac{\sigma}{\|L\|}\|(-(t^* - t_i)L + I)Lx(t_i)\|.$$

Using the notation $\xi = t^* - t_i$, the latter is rewritten as

$$\begin{aligned}\|Lx(t_i)\|^2\|L\|^2\xi^2 = \sigma^2(&\|L^2x(t_i)\|^2\xi^2 \\ &+ \|L^2x(t_i)\|^2 - 2(Lx(t_i))^T LLx(t_i)\xi),\end{aligned}$$

or equivalently,

$$\begin{aligned}(\|Lx(t_i)\|^2\|L\|^2 - \sigma^2\|L^2x(t_i)\|^2)\xi^2 \\ + 2\sigma^2(Lx(t_i))^T LLx(t_i)\xi - \sigma^2\|L^2x(t_i)\|^2 = 0.\end{aligned}$$

Note that

$$(\|Lx(t_i)\|^2\|L\|^2-\sigma^2\|L^2x(t_i)\|^2) > (1-\sigma^2)\|Lx(t_i)\|^2\|L\|^2,$$

so that $(\|Lx(t_i)\|^2\|L\|^2 - \sigma^2\|L^2x(t_i)\|^2) > 0$ and

$$\begin{aligned}\Delta =&4\sigma^4\|(Lx(t_i))^T LLx(t_i)\|^2 + 4\sigma^2\|L^2x(t_i)\|^2\\ &\cdot (\|Lx(t_i)\|^2\|L\|^2 - \sigma^2\|L^2x(t_i)\|^2) > 0.\end{aligned}$$

An upper bound is then given by

$$t^* = t_i + \frac{-2\sigma^2(Lx(t_i))^T LLx(t_i) + \sqrt{\Delta}}{2(\|Lx(t_i)\|^2\|L\|^2 - \sigma^2\|L^2x(t_i)\|^2)}. \tag{10}$$

Note that as long as $Lx(t_i) \neq 0$, i.e., agreement has not been reached, $t^* - t_i$ is strictly positive, i.e., the inter-execution times are non-trivial. The preceding analysis, along with Theorem 1, yield the following result:

Theorem 2: Consider system $\dot{x} = u$ with the control law (6) and assume that the communication graph G is connected. Suppose that $0 < \sigma < 1$. Assume that for each $i = 1, 2, \ldots$ the next update time is chosen such that the bound

$$t_{i+1} - t_i < \frac{-2\sigma^2(Lx(t_i))^T LLx(t_i) + \sqrt{\Delta}}{2(\|Lx(t_i)\|^2\|L\|^2 - \sigma^2\|L^2x(t_i)\|^2)} \tag{11}$$

holds. Then for any initial condition in $\mathbb{R}^N$ all agents converge to their initial average, i.e.,

$$\lim_{t\to\infty} x_i(t) = \bar{x} = \frac{1}{N}\sum_i x_i(0), \quad \forall i \in \mathcal{N}.$$

IV. Distributed Self-triggered Control

A. Review of Distributed Event-Triggered Control Design

In this section, we consider a distributed counterpart of the event-triggered agreement problem. In particular, each agent now updates its own control input at event times it decides based on information from its neighboring agents. The event times for each agent $i \in \mathcal{N}$ are denoted by $t_0^i, t_1^i, \ldots$. We will first review the event-triggered approach of [7] and proceed to the self-triggered formulation in the sequel.

The measurement error for agent i is defined as

$$e_i(t) = x_i(t_k^i) - x_i(t), t \in [t_k^i, t_{k+1}^i). \tag{12}$$

The distributed control law for agent i is now given by:

$$u_i(t) = -\sum_{j\in N_i}\Big(x_i(t_k^i) - x_j(t^j_{k'(t)})\Big), \tag{13}$$

where $k'(t) \triangleq \arg\min_{l\in\mathbb{N}:t\geq t_l^j}\Big\{t - t_l^j\Big\}$. Hence, each agent takes into account the last update value of each of its neighbors in its control law. The control law for i is updated both at its own event times $t_0^i, t_1^i, \ldots$, as well as at the event times of its neighbors $t_0^j, t_1^j, \ldots, j \in N_i$. It is shown in [7] that in this case we also have $\dot{\bar{x}} = 0$ for the agents' initial average.

Denote now $Lx \triangleq z = [z_1, \ldots, z_N]^T$ and consider $V = \frac{1}{2}x^T Lx$.. Then it is shown in [7] that

$$\begin{aligned}\dot{V} \leq& -\sum_i z_i^2 + \sum_i a|N_i|z_i^2\\ &+\sum_i \frac{1}{2a}|N_i|e_i^2 + \sum_i\sum_{j\in N_i}\frac{1}{2a}e_j^2,\end{aligned}$$

for $a > 0$.

Since the graph is symmetric, by interchanging the indices of the last term we get

$$\sum_i\sum_{j\in N_i}\frac{1}{2a}e_j^2 = \sum_i\sum_{j\in N_i}\frac{1}{2a}e_i^2 = \sum_i\frac{1}{2a}|N_i|e_i^2,$$

so that $\dot{V} \leq -\sum\limits_i (1 - a|N_i|)z_i^2 + \sum\limits_i \frac{1}{a}|N_i|e_i^2$. Assume that a satisfies $0 < a < \frac{1}{|N_i|}$ for all $i \in \mathcal{N}$. Then, enforcing the condition

$$e_i^2 \leq \frac{\sigma_i a(1 - a|N_i|)}{|N_i|}z_i^2, \tag{14}$$

we get $\dot{V} \leq \sum\limits_i (\sigma_i - 1)(1 - a|N_i|)z_i^2$, which is negative definite for $0 < \sigma_i < 1$.

Thus for each i, the event times are defined recursively by

$$t_{k+1}^i = \arg\min_t\{t : e_i^2(t) = \frac{\sigma_i a(1 - a|N_i|)}{|N_i|}z_i^2(t), t \geq t_k^i\}, \tag{15}$$

with $t_0^i = 0$ and where $z_i = \sum\limits_{j\in N_i}(x_i - x_j)$. The main result of [7] is summarized in the following:

Theorem 3: Consider the system $\dot{x} = u$ with the control law (13), (15) and assume that the communication graph G is connected. Suppose that $0 < \sigma < 1$ and $0 < a < \frac{1}{|N_i|}$. Then the states of all agents converge to their initial average, i.e., $\lim_{t\to\infty} x_i(t) = \bar{x} = \frac{1}{N}\sum\limits_i x_i(0)$ for all $i \in \mathcal{N}$.

B. Distributed Self-Triggered Control

Similarly to the centralized case, continuous monitoring of the measurement error norm is required to check condition (15) in the distributed case. In the self-triggered setup, the next time t_{k+1}^i at which control law is updated is predetermined at the previous event time t_k^i and no state or error measurement is required in between the control updates. Such a distributed self-triggered control design is presented below.

Define

$$\beta_i = \frac{\sigma_i a(1 - a|N_i|)}{|N_i|}.$$

Then, (14) is rewritten as

$$|x_i(t_k^i) - x_i(t)|^2 \leq \beta_i z_i^2(t).$$

Since

$$\dot{x}_i(t) = -\sum_{j\in N_i}\Big(x_i(t_k^i) - x_j(t_{k'}^j)\Big),$$

we get

$$x_i(t) = -\sum_{j\in N_i}(x_i(t_k^i) - x_j(t_{k'}^j))(t - t_k^i) + x_i(t_k^i)$$

for $t \in [t_k^i, \min\{t_{k+1}^i, \min_{j\in N_i} t_{k''}^j\})$, where

$$k'' \triangleq \arg \min_{l\in\mathbb{N}: t_k^i \le t_l^j} \left\{ t_l^j - t_k^i \right\}$$

and hence $\min\{t_{k+1}^i, \min_{j\in N_i} t_{k''}^j\}$ is the next time when the control u_i is updated. Thus (14) is equivalent to

$$|\sum_{j\in N_i} (x_i(t_k^i) - x_j(t_{k'}^j))(t - t_k^i)|^2 \le \beta_i z_i^2(t). \tag{16}$$

Recalling

$$z_i(t) = \sum_{j\in N_i} (x_i(t) - x_j(t)),$$

we also have

$$x_j(t) = -\sum_{l\in N_j} (x_j(t_{k'}^j) - x_l(t_{k'''}^l))(t - t_{k'}^j) + x_j(t_{k'}^j),$$

where

$$k''' = k'''(t) \triangleq \arg \min_{m\in\mathbb{N}: t\ge t_m^l} \left\{ t - t_m^l \right\}.$$

Denote now

$$\sum_{j\in N_i} (x_i(t_k^i) - x_j(t_{k'}^j)) = \rho_i, \sum_{l\in N_j} (x_j(t_{k'}^j) - x_l(t_{k'''}^l)) = \rho_j,$$

and

$$\xi_i = t^* - t_k^i.$$

We can compute

$$\begin{aligned} z_i(t) &= \sum_{j\in N_i} (x_i(t) - x_j(t)) \\ &= \sum_{j\in N_i} (-\rho_i \xi_i + x_i(t_k^i)) \\ &- \sum_{j\in N_i} (-\rho_j (t - t_{k'}^j) + x_j(t_{k'}^j)) \\ &= -|N_i|\rho_i \xi_i + |N_i| x_i(t_k^i) \\ &+ \sum_{j\in N_i} (\rho_j (t - t_k^i + t_k^i - t_{k'}^j) - x_j(t_{k'}^j)), \end{aligned}$$

or equivalently,

$$\begin{aligned} z_i(t) = (-|N_i|\rho_i + &\sum_{j\in N_i} \rho_j)\xi_i \\ &+\rho_i + \sum_{j\in N_i} (\rho_j (t_k^i - t_{k'}^j)). \end{aligned}$$

Further denoting $P_i = -|N_i|\rho_i + \sum_{j\in N_i} \rho_j$ and $\Phi_i = \rho_i + \sum_{j\in N_i} (\rho_j(t_k^i - t_{k'}^j))$, the condition (16) can be rewritten as $|\rho_i \xi_i| \le \sqrt{\beta_i} |P_i \xi_i + \Phi_i|$ and since $\xi_i \ge 0$, the latter is equivalent to

$$|\rho_i| \xi_i \le \sqrt{\beta_i} |P_i \xi_i + \Phi_i|. \tag{17}$$

Note that this inequality always holds for $\xi_i = 0$. Also note that (16) may or may not hold for all $\xi_i \ge 0$, and this can be decided by agent i at time t_k^i. Based on this observation, the self-triggered policy for agent i at time t_k^i is defined as follows: if there is a $\xi_i \ge 0$ such that $|\rho_i|\xi_i = \sqrt{\beta_i}|P_i\xi_i + \Phi_i|$ then the next update time t_{k+1}^i takes place at most ξ_i time units after t_k^i, i.e., $t_{k+1}^i \le t^* = t_k^i + \xi_i$. Of course if there is an update in one of its neighbors, thus updating the control law (13), then agent i re-checks the condition. Otherwise, if the inequality $|\rho_i|\xi_i \le \sqrt{\beta_i}|P_i\xi_i + \Phi_i|$ holds for all $\xi_i \ge 0$, then agent i waits until the next update of the control law of one of its neighbors to re-compute this condition. Note that in [7], we showed that there is a strictly positive solution $\xi_i > 0$ for at least one i at each time instant.

The self-triggered ruling for each agent i is thus summarized as:

Definition 4: For each $i = 1, 2, \ldots$ the self-triggered ruling defines the next update time as follows: if there is a $\xi_i \ge 0$ such that $|\rho_i|\xi_i = \sqrt{\beta_i}|P_i\xi_i + \Phi_i|$, then the next update time t_{k+1}^i takes place at most ξ_i time units after t_k^i, i.e., $t_{k+1}^i \le t^* = t_k^i + \xi_i$. Agent i also checks this condition whenever its control law is updated due an update of the error of one of its neighbors. Otherwise, if the inequality $|\rho_i|\xi_i \le \sqrt{\beta_i}|P_i\xi_i + \Phi_i|$ holds for all $\xi_i \ge 0$, then agent i waits until the next update of the control law of one of its neighbors to re-check this condition.

The preceding analysis, along with Theorem 3, yield the following result:

Theorem 5: Consider system $\dot{x} = u$ with the control law (13) and assume that the communication graph G is connected. Suppose that $0 < a < \frac{1}{|N_i|}$ and $0 < \sigma_i < 1$ for all $i \in \mathcal{N}$. Assume that for each $i = 1, 2, \ldots$ the next update time is decided according to Definition 4.

Then, for any initial condition in $\mathbb{R}^N$, the states of all agents converge to their initial average, i.e.,

$$\lim_{t\to\infty} x_i(t) = \bar{x} = \frac{1}{N} \sum_i x_i(0),$$

for all $i \in \mathcal{N}$.

The previous analysis can also help us derive some conclusions about the inter-execution times of each agent.

Note that after simple calculation it is easily derived that $\Phi_i = z_i(t_k^i)$. From (17), we know that the next event for agent i occurs at a time t when the equation

$$|\rho_i|(t - t_k^i) \le \sqrt{\beta_i}|P_i(t - t_k^i) + z_i(t_k^i)|$$

holds. Thus a zero inter-execution time for agent i can only occur when $|z_i(t_k^i)| = 0$. By virtue of Theorem 5, the system is asymptotically stabilized to the initial average. By the Cauchy-Schwartz inequality, we have

$$||z||^2 = ||Lx||^2 = |\sum_i \sum_{j\in N_i} (x_i - x_j)|^2 \le \frac{1}{2} x^T L x = V$$

so that z asymptotically converges to zero. Unfortunately there is no guarantee that no element of z will reach zero in finite time (or be equal to zero initially), however, as shown above, the inter-execution time can only be zero when $z_i = 0$ for agent i, i.e., when agent i has already reached its control objective.

V. EXAMPLES

The results of the previous sections are illustrated through computer simulations. In the following paragraphs, we consider both the centralized and distributed formulations of the self-triggered algorithms and compare the derived results with the corresponding event-triggered formulation of [7].

As in [7], consider a network of four agents whose Laplacian matrix is given by

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 3 & -1 & -1 \\ 0 & -1 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

The four agents start from random initial conditions and evolve under the control law (6) in the centralized case, and the control law (13) in the distributed case. In the centralized case, we have set $\sigma = 0.65$, and $\sigma_1 = \sigma_2 = 0.55$, $\sigma_3 = \sigma_4 = 0.75$ and $a = 0.2$ for the distributed control example. In both cases, we consider two different cases of actuation updates: the event-triggered and the self-triggered one.

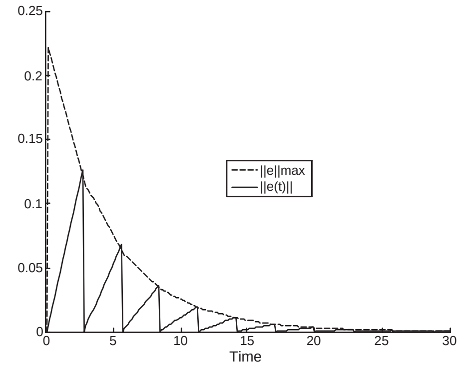
Figure 1 shows the evolution of the error norm in the centralized case. The top plot represents the event-triggered and the bottom the self-triggered formulation. In the event-triggered case, the control law is updated according to Theorem 1 and in the self-triggered according to Theorem 2. The solid line represents the evolution of the error $||e(t)||$. This stays in both plots below the specified state-dependent threshold $||e||_{max} = \sigma \frac{\|Lx\|}{\|L\|}$ which is represented by the dotted line in the Figure.

The next simulation depicts how the framework is realized in the distributed case for agent 1. In particular, the solid line in Figure 2 shows the evolution of $|e_1(t)|$. This stays below the specified state-dependent threshold given by (14) $|e_1|_{\max} = \sqrt{\frac{\sigma_1 a(1-a|N_1|)}{|N_1|}} z_1$ which is represented by the dotted line in the Figure. Once again, the top plot shows the event-triggered case of Theorem 3 and the bottom plot the self-triggered case of Theorem 5.

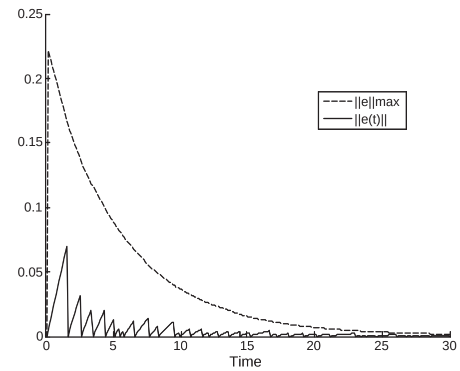
In both cases, it can be seen that the event-triggered case requires less controller updates. On the other hand, the seld triggered approach seems more robust, since the design provides an upper bound on the interval in which the update should be held.

VI. CONCLUSIONS

We presented a self-triggered control strategy for a multi-agent system with single integrator agents. This approach extends our results reported previously for event-triggered multi-agent control to a self-triggered framework, where each agent now computes its next update time at the previous one, without having to keep track of the state error that triggers the actuation between two consecutive update instants. The approach was presented both from a centralized and a distributed perspective.



(a) Event-triggered case



(b) Self-triggered case

Fig. 1. Four agents evolve under the centralized event-triggered (top plot) and self-triggered (bottom plot) proposed framework.

Future work will involve extending the proposed approach to more general dynamic models, as well as adding uncertainty and time delays to the information exchange.

REFERENCES

[1] A.Anta and P.Tabuada. To sample or not to sample: self-triggered control for nonlinear systems. *IEEE Transactions on Automatic Control*, 2009. to appear.

[2] M. Arcak. Passivity as a design tool for group coordination. *IEEE Transactions on Automatic Control*, 52(8):1380–1390, 2007.

[3] A. Arsie and E. Frazzoli. Efficient routing of multiple vehicles with no communications. *International Journal of Robust and Nonlinear Control*, 18(2):154–164, 2007.

[4] K.J. Astrom and B. Bernhardsson. Comparison of Riemann and Lebesgue sampling for first order stochastic systems. *41st IEEE Conference on Decision and Control*, pages 2011–2016, 2002.

[5] M. Cao, B.D.O. Anderson, A.S. Morse, and C. Yu. Control of acyclic formations of mobile autonomous agents. *47th IEEE Conference on Decision and Control*, pages 1187–1192, 2008.

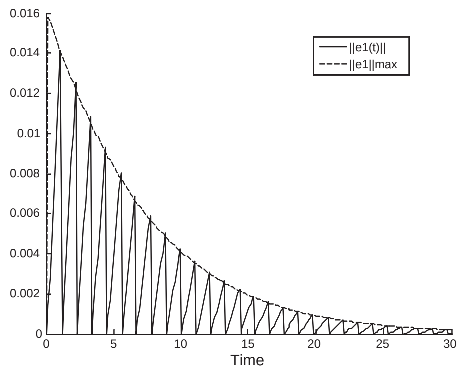
[6] L. Consolini, F. Morbidi, D. Prattichizzo, and M. Tosques. Leader-follower formation control of nonholonomic mobile robots with input constraints. *Automatica*, 44(5):1343–1349, 2008.

[7] D.V. Dimarogonas and K.H. Johansson. Event-triggered control for multi-agent systems. *48th IEEE Conf. Decision and Control*, 2009. to appear.

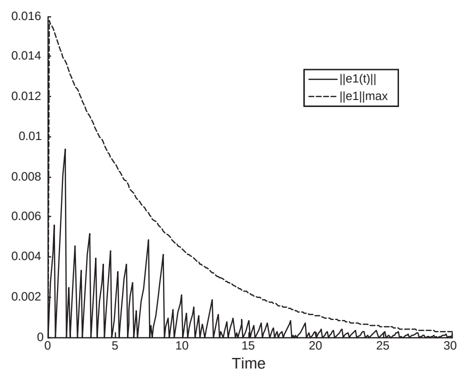
[8] D.V. Dimarogonas and K.J. Kyriakopoulos. A connection between formation infeasibility and velocity alignment in kinematic multi-agent systems. *Automatica*, 44(10):2648–2654, 2008.

[9] J.A. Fax and R.M. Murray. Graph Laplacians and stabilization of vehicle formations. *15th IFAC World Congress*, 2002.

[10] C. Godsil and G. Royle. *Algebraic Graph Theory*. Springer Graduate Texts in Mathematics # 207, 2001.



(a) Distributed event-triggered case



(b) Distributed self-triggered case

Fig. 2. Four agents evolve under the distributed event-triggered (top plot) and self-triggered (bottom plot) proposed framework.

[11] L. Grüne and F. Müller. An algorithm for event-based optimal feedback control. *48th IEEE Conf. Decision and Control*, pages 5311–5316, 2009.

[12] M. Haque and M. Egerstedt. Decentralized formation selection mechanisms inspired by foraging bottlenose dolphins. *Mathematical Theory of Networks and Systems*, 2008.

[13] W.P.M.H. Heemels, J.H. Sandee, and P.P.J. Van Den Bosch. Analysis of event-driven controllers for linear systems. *International Journal of Control*, 81(4):571–590, 2007.

[14] E. Johannesson, T. Henningsson, and A. Cervin. Sporadic control of first-order linear stochastic systems. *Hybrid Systems: Computation and Control*, pages 301–314, 2007.

[15] D. Lehmann and J. Lunze. Event-based control: A state feedback approach. *European Control Conference*, pages 1716–1721, 2009.

[16] S.G. Loizou and K.J Kyriakopoulos. Navigation of multiple kinematically constrained robots. *IEEE Transactions on Robotics*, 2008. to appear.

[17] M. Mazo, A. Anta, and P. Tabuada. On self-triggered control for linear systems: Guarantees and complexity. *European Control Conference*, 2009.

[18] M. Mazo and P. Tabuada. On event-triggered and self-triggered control over sensor/actuator networks. *47th IEEE Conf. Decision and Control*, pages 435–440, 2008.

[19] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[20] R. Olfati-Saber and J.S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. *44th IEEE Conference on Decision and Control*, pages 6698–6703, 2005.

[21] M. Rabi, K.H. Johansson, and M. Johansson. Optimal stopping for event-triggered sensing and actuation. *47th IEEE Conference on Decision and Control*, pages 3607–3612, 2008.

[22] W. Ren and E.M. Atkins. Distributed multi-vehicle coordinated control via local information exchange. *International Journal of Robust and Nonlinear Control*, 17(10-11):1002–1033, 2007.

[23] A. Speranzon, C. Fischione, and K.H. Johansson. Distributed and collaborative estimation over wireless sensor networks. *45th IEEE Conference on Decision and Control*, pages 1025–1030, 2006.

[24] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[25] M. Velasco, J. Fuertes, and P. Marti. The self-triggered task model for real-time control systems. *In Work in Progress Proceedings of the 24th IEEE Real-Time Systems Symposium*, pages 67–70, 2005.

[26] X. Wang and M.D. Lemmon. Event-triggered broadcasting across distributed networked control systems. *American Control Conference*, 2008.

[27] X. Wang and M.D. Lemmon. Self-triggered feedback control systems with finite-gain L2 stability. *IEEE Transactions on Automatic Control*, 2008. To appear.

[28] F. Xie and R. Fierro. On motion coordination of multiple vehicles with nonholonomic constraints. *American Control Conference*, pages 1888–1893, 2007.